Supplementary Methods

Supplementary Method 1: The Cancer Genome Atlas (TCGA) Data Acquisition and Cohort Definition

TCGA Data Acquisition and Preprocessing

Publicly available raw gene expression data for the Cervical Squamous Cell Carcinoma and Endocervical Adenocarcinoma (TCGA-CESC) project were retrieved using the *TCGAbiolinks* R package using the following parameters: project = "TCGA-CESC", data.category = "Transcriptome Profiling", data.type = "Gene Expression Quantification", and workflow.type = "STAR - Counts". Raw count data were downloaded and prepared into a *SummarizedExperiment* object using GDCprepare.

To generate a gene-level expression matrix, raw counts were extracted, and Ensembl gene identifiers (ENSG) were trimmed to remove version numbers (e.g., ENSG000001.5 converted to ENSG000001). Gene symbol annotation was performed using the *biomaRt* R package by querying the hsapiens_gene_ensembl dataset to map Ensembl IDs to HGNC external gene names. Data were aggregated by retaining the maximum expression value across all associated identifiers for each sample to address redundant Ensembl IDs. Entries without a valid gene symbol mapping were discarded.

Clinical Cohort Selection

Corresponding clinical and survival phenotypes were obtained from the UCSC Xena platform (files: "CESC_survival.txt" and "CESC_clinicalMatrix.txt") and merged with genomic data using the unique patient identifier (_PATIENT). Patients were included if they met the following criteria:

1) **Locally Advanced Disease:** Diagnosis of FIGO 2009 stage IB2, II, IIB, III, IIIA, IIIB, or IVA, as defined by the clinical_stage variable.

2) **Sample Type:** Analysis was restricted to samples annotated as "Primary Tumor" via the sample_type variable.

3) **Chemoradiation:** Documented receipt of radiation therapy (radiation_therapy = "YES"). Patients were excluded if they underwent surgery, as defined in hysterectomy_performed_type or hysterectomy_performed_text variables.

4) **Treatment Completion:** Patients where external_beam_radiation_therapy_administered_status was recorded as "Treatment not completed." were excluded.

5) **Chemotherapy:** Patients were excluded if chemotherapy_regimen_type was missing, empty, or contained ambiguous annotations (e.g., "Cisplatin|Carboplatin|Other|Other"). The primary clinical

endpoint, treatment response, was derived from the primary_therapy_outcome_success variable (Complete Remission/Response"; n = 27, "Partial Remission/Response" n = 3; "Stable Disease" n = 2; and "Progressive Disease" n = 6). Patients with a recorded outcome were selected and analysed as combined "Partial Remission/Response" and "Stable Disease" vs. "Complete Remission/Response".

Differential Gene Expression

Raw count data was filtered to retain genes with at least 10 counts in a minimum number of samples equal to the smallest group size. Analysis was restricted to genes encoding proteins detectable in plasma. Differential expression between combined "Partial Remission/Response" and "Stable Disease" vs. "Complete Remission/Response" groups was assessed using a negative binomial generalized linear model by DESeq2 R package. Log2 fold change estimates were adjusted using the adaptive shrinkage estimator to account for effect size accuracy of genes with low counts or high dispersion. Genes were considered differentially expressed if they met both an BH adjusted p-value threshold of <0.05 and fold change ≥ 1.2 in either direction.

Supplementary Method 2: NetworkAnalyst parameters

PPI networks were constructed using the generic IMEx Interactome database supplied in NetworkAnalyst interface, which contains exclusively generic experimentally validated interactions curated to IMEx consortium standards, with no user-adjustable confidence thresholds implemented. First-order networks were built by expanding from seed genes (union list of differentially expressed genes/protein: PR/SD vs. CR from PRM dataset and combined Partial Remission/Response and Stable Disease vs. Complete Remission/Response from TCGA dataset; no direction) using direct first-order interactions. The final output was limited to only nodes with more than one degree and betweenness more than zero (i.e. non-terminal node or dead node). KEGG enrichment was done on the remaining nodes.

Supplementary Method 3: PRM Feature Selection and Ridge Regression Modelling

Data Preprocessing and Predictive Model Development

The machine learning model pipeline was used to identify protein that discriminate PR/SD vs. CR from PRM data. Protein intensity values were $\log_2$-transformed prior to analysis. Missing data was imputed by minimal probabilistic procedure. The standardization of predictors was performed internally within glmnet (mean-centering and scaled), prior to penalty application. Prediction model was done with ridge regression model, with LASSO and elastic-net as supplementary procedure. The model was implemented as follows:

The complete modeling pipeline was applied to the full dataset to obtain apparent performance. This pipeline consisted of three steps:

1) Feature selection using two-sample t-tests with Benjamini-Hochberg false discovery rate correction at a threshold of 0.05.

2) Selection of the regularization parameter λ via 10-fold cross-validation using the one-standard-error rule (lambda.1se), which selects the most parsimonious model within one standard error of the minimum cross-validation error.

3) Fitting of the regression model using the selected features and regularization parameter. The area under the receiver operating characteristic curve (AUC-ROC) computed on the full dataset constituted the apparent AUC.

Bootstrap resampling was performed with 2,000 iterations. In each iteration, a bootstrap sample of size n was drawn with replacement from the original dataset, where n equals the original sample size. The entire modeling pipeline, including both t-tests for feature selection and cross-validation for λ selection was performed within the bootstrap sample. The number of cross-validation folds was set to the minimum of 10 or the smallest class size in the bootstrap sample. After fitting the final model on the bootstrap sample, model performance was assessed using Harrell's optimism correction as follows: the AUC on the bootstrap sample itself (AUC_boot) and the AUC obtained by applying the bootstrap-derived model to the original dataset (AUC_orig). The optimism for each iteration was calculated as the difference between these values (Optimism = AUC_boot − AUC_orig).

The optimism-corrected AUC was computed by subtracting the mean optimism across all valid bootstrap iterations from the apparent AUC. Next, 95% confidence intervals were derived from the 2.5th and 97.5th percentiles of the distribution of AUC values obtained by applying each bootstrap model to the original data. Bootstrap iterations were considered valid only if both outcome classes contained at least three observations, at least two proteins passed the feature selection threshold, and model fitting converged successfully.

Protein Importance and Stability Assessment

To evaluate the robustness of individual protein contributions to the predictive model, we computed selection stability metrics across bootstrap iterations as follows:

1) Inclusion frequency (proportion of iterations that specific proteins were included in the model).

2) Mean coefficient value when included.

3) Sign consistency (proportion of iterations with the same coefficient sign).

4) Stability score = Inclusion frequency * |mean coefficient|.